EXPRESS MAIL CERTIFICATE

Date: 2/9/00 Label No. EL284833151US

I hereby certify that, on the date indicated above I deposited this paper or fee with the U.S. Postal Service and that it was addressed for delivery to Box Patent Application, the Assistant Commissioner for Patents, Washington, DC 20231 by "Express Mail Post Office to Addressee" service.

ROSEMARIE MEDINA Rosemarie Medina
Name (Print) Signature

APPLICATION
FOR
UNITED STATES LETTERS PATENT

TO ALL WHOM IT MAY CONCERN:

Be it known that Michael Pothier, John Lynch, David Cairns, Glenn Eng, and Andre Cyr have invented a System and Method for Synchronizing with Data Received Over an Unreliable Asynchronous Medium, of which the following description in connection with the accompanying drawings is a specification, like reference characters on the drawings indicating like parts in the several Figures.

SYSTEM AND METHOD FOR SYNCHRONIZING WITH DATA RECEIVED OVER AN UNRELIABLE ASYNCHRONOUS MEDIUM

FIELD OF THE INVENTION

This invention relates generally to the transmission of synchronous data by means of an unreliable asynchronous connection, and more particularly to re-establishing synchronization with the data after reception from the asynchronous connection.

BACKGROUND OF THE INVENTION

At a location such as the offices of a business, there typically are many telephones that may be used to communicate among themselves as well as over the public switched telephone network (PSTN) with other telephones at other locations. In such an application, it is typical practice to provide a PBX at the office location for facilitating the connections among the telephones at the office location.

A basic PBX installation consists of a cabinet to which a number of phones may be connected, and which contains the circuitry for interconnecting those phones. The number of phones that may be serviced by one cabinet is limited by the amount of circuitry that may be accommodated within the cabinet. A typical number of phones serviced by one PBX cabinet is on the order of 50 to 100.

If it is desired to service a number of phones greater than that which may be serviced by one PBX cabinet, the PBX may be augmented by additional PBX cabinets connected to the original one. Fig. 1 depicts two PBX cabinets 101, each servicing a number of phones. The two PBX cabinets 101 are connected by a wire path 102, through which a phone connected to one of the PBX cabinets may communicate with a phone connected to another of the PBX cabinets. Path 102 may be other forms of hard path other than a wire path, such as a fiber optic path. All the cabinets comprising a single PBX are in the same general location, typically no more than several hundred feet apart. By using repeaters, the distance between cabinets can be extended to several miles.

As the distance between PBX cabinets increases, (e.g., when interconnecting phones in different buildings on an office campus) it may be convenient to interconnect the PBX cabinets by means other than a copper or fiber optic path. For example, it may be convenient to interconnect PBX cabinets by an Internet Protocol (IP) connection. An IP network may already be in existence. However, when interconnecting PBX cabinets it is necessary to transmit not only data but also a timing reference. Copper path 102 (Fig. 1) carries a timing reference. However, an IP connection (e.g., an Ethernet connection) transmits data asynchronously, i.e., without a timing reference. Also, an IP network routes each packet according to current network conditions, and thus as conditions change packets may take different routes and may be held up for varying amounts of time. When an IP network is lightly loaded and functioning under optimal conditions, it probably delivers output at substantially the same rate as its input, but under adverse conditions, such as heavy load, packets may be delayed by markedly different amounts of time, and may even be lost entirely. Packets may thus arrive at the destination at a different instantaneous rate and in a different order than that in which they were transmitted, may arrive too late to be acted upon, or may not arrive at all.

There is thus a need to reconstruct a timing reference for synchronous data after transmission thereof by an asynchronous, unreliable communication medium which may deliver data out of order, which may deliver data at an instantaneous rate substantially different than its transmission rate, and which may lose portions of the data.

It is an advantage of the present invention that a system and method is provided for reconstructing a timing reference for synchronous data that has been transmitted by an asynchronous communication medium.

It is a further advantage of the present invention that a system and method is provided for reconstructing such a timing reference for synchronous data that has been transmitted out of order and with data loss by an unreliable asynchronous communication medium.

These and other advantages of the invention will become apparent to those skilled in the art from the following description thereof.

SUMMARY OF THE INVENTION

In accordance with the teachings of the present invention, these and other advantages may be accomplished by the present systems and methods of reestablishing synchrony with formerly synchronous data that has been transmitted over an asynchronous network.

An embodiment of the present invention includes synchronizing a clock with data received via an asynchronous transmission medium by holding in buffers blocks of data received from the transmission medium, reading out the data from the buffers at a rate specified by the clock, and regulating the rate specified by the clock according to occurrence of data in the buffers.

An aspect of the present invention includes regulating the rate specified by the clock so that the data is read out from the buffers at a rate substantially equal to a rate at which the data is currently arriving via the asynchronous transmission medium.

Another aspect of the invention includes initially setting the rate specified by the clock to a predetermined constant rate at which data is being input to the asynchronous transmission medium.

Another aspect of the invention includes regulating the clock rate so the point of reading out from the buffers stays a predetermined distance from the point of storing into the buffers.

Another aspect of the invention includes transmitting the data as packets, determining whether a packet has been lost in transmission, and reconstituting the packet at the destination with arbitrary data so that timing relationships among the other packets are not affected.

The invention will next be described in connection with certain exemplary embodiments; however, it should be clear to those skilled in the art that various modifications, additions and subtractions can be made without departing from the spirit or scope of the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be more clearly understood by reference to the following detailed description of an exemplary embodiment in conjunction with the accompanying drawings, in which:

Fig. 1 (prior art) depicts two PBX cabinets, each servicing a number of phones, interconnected by a copper path;

Fig. 2 depicts two PBX cabinets, each servicing a number of phones, interconnected by an Ethernet/IP network;

Fig. 3 is a timing diagram for assembling synchronous PBX data into Ethernet/IP packets according to the present invention;

Figs. 4A and 4B show the structure of Ethernet/IP packets carrying synchronous PBX data according to the present invention;

Fig. 5 illustrates the buffering of received Ethernet/IP packets according to the present invention;

Fig. 6 is a block diagram of a system for reconstructing a synchronous clock after asynchronous transmission according to the present invention; and

Fig. 7 is a flow chart of an alternative method for reconstructing a synchronous clock after asynchronous transmission according to the present invention;

Fig. 8 is a flow chart of an alternative method for reconstructing a synchronous clock after asynchronous transmission according to the present invention;

Fig. 9 is a flow chart of an alternative method for reconstructing a synchronous clock after asynchronous transmission according to the present invention;

Fig. 10 is a block diagram of a system for reconstructing a synchronous clock after asynchronous transmission according to the present invention.

DESCRIPTION OF THE INVENTION

Fig. 2 differs from Fig. 1 in that Fig. 2 shows two PBX cabinets 101 connected by an asynchronous communication network 103 (e.g., an Ethernet/IP network in a present embodiment), whereas Fig. 1 depicts the conventional copper path 102 for connecting two PBX cabinets. In applications where the phones to be serviced by the various PBX cabinets 101 are spread over a wide area (as in several buildings on an office campus) it may be more convenient or economic to use something other than copper path 102. Although the present embodiment uses Ethernet/IP network 103 to connect the PBX cabinets 101, those skilled in the art appreciate that other forms of asynchronous connection, such as asynchronous wireless connection, etc., might be employed.

A present embodiment is based on a model DS30 PBX marketed by Nortel Networks. The ensuing discussion relates specifics (timing, numbers of channels, etc.) unique to this device, but those skilled in the art appreciate that many of these specifics are design choices that may be departed from without departing from the scope and spirit of the invention. Those skilled in the art also appreciate that the present invention is not limited to use in a PBX switch but could be employed in many different circuit switches.

Fig. 3 depicts timing relationships pertaining to the occurrences of data within a PBX cabinet 101 and the packaging of data into Ethernet/IP packets for transmission over Ethernet/IP network 103 to another PBX cabinet 101. The ensuing discussion considers one PBX cabinet 101 to be a source and another to be a destination. Those skilled in the art realize that two-way communication is involved, and thus each PBX functions in both roles.

A PBX cabinet 101 in the present embodiment contains one or more digital line cards (DLC) (not shown) which can each service 16 phones. Each phone in the present embodiment can have two channels, a voice channel and a data channel. There are thus 32 channels serviced by a DLC.

Line A on Fig. 3 shows that there is a 1-KHz. clock pulse generated within PBX cabinet 101 which establishes a 1000-μsec. clock period. The 1000-μsec clock period is further divided into eight 125-μsec. frames delineated by an 8-KHz. clock and connoted 1 – 8 (line B). As indicated on lines C01 – C32, 10 bits of data are established for each channel during each frame. As shown in element 410 in Fig. 4A, the 10 bits for each channel comprise an 8-bit PCM sample of the audio emanating from a microphone of the phone if a voice channel or 8 bits of data if a data channel, a signaling bit S, and a signaling validation bit V. In the present embodiment, a phone may send a control signal coded in an 8-bit code comprising the S bits of the eight ten-bit groups in one clock period. (Control signals are used to indicate such things as "the phone has been taken off hook", "the asterisk key is being depressed", etc.) While a control signal is active (as opposed to idle condition of the S bit) the V bit is on as well.

Fig. 3 shows that the data accumulated from the 32 channels during a 500-μsec. period is packaged into an Ethernet/IP packet, with alternate 500-μsec. periods coinciding with the start of a 1000-μsec. clock period. Channel data obtained during frames 1-4 of a clock period are packaged as shown in Fig. 4A, while channel data obtained during frames 5-8 of a clock period are packaged as shown in Fig. 4B.

Figs. 4A and 4B show that an Ethernet/IP packet comprises an Ethernet header 402, an Internet Protocol (IP) header 404, and a payload 406 or 408. The headers 402 and 404 are standard headers. The Frame Check Sequence (FCS) 407 is as specified in Ethernet Standards 802.3. The payloads 406 and 408 are identical in form but vary in content. Each payload begins with a sequence (SEQ) field which contains a sequence value for distinguishing a packet from its neighbors. In one embodiment the SEQ field is transmitted with alternating values chosen from two values, which distinguishes a packet from its immediate predecessor and its immediate successor. In other embodiments the SEQ field contains values chosen from a larger sequence of values which enables distinction from a larger number of neighboring packets. For example, the first two-hundred packets sent might have SEQ fields of 0 through 199 respectively, the second two-hundred packets sent might again have SEQ fields of 0 through 199 respectively, and so on.

Payload 406 then contains the ten-bit values 410 for each of the 32 channels for each of the frames 1 – 4, in the order: channel 1 through channel 32 for frame 1; channel 1 through channel 32 for frame 2; channel 1 through channel 32 for frame 3; and channel 1 through channel 32 for frame 4. Payload 408 is in corresponding order, but for frames 5, 6, 7, and 8.

Although a packet is transmitted every 500 μsec in the present embodiment, the duration of each packet is not necessarily equal to 500 μsec, and is probably less. Since various ones of the 32 channels depicted in Fig. 3 may become inactive at various times, the duration of the packets is variable. The packet transmission interval, however, is constant at 500 μsec in the present embodiment.

The Ethernet/IP packets as detailed in Figs. 4A and 4B are then transmitted via Ethernet/IP network 103 to another PBX cabinet 101. Unlike conventional transmission over a copper path 102 (Fig. 1) or some alternative path which permits synchronous transmission, transmission via Ethernet/IP is inherently asynchronous. There is no provision for including any of the clock signals in the Ethernet/IP packets. Their inclusion, if attempted, would be meaningless since Ethernet transmission is inherently asynchronous and the arrival of clock signals at the destination is subject to indeterminate delay and variation. In order to play back the PCM and signaling data correctly at a destination PBX cabinet 101, suitable timing is reconstructed at the destination PBX cabinet 101.

Fig. 5 shows buffering provided at a destination PBX cabinet 101. Buffers 500 have sufficient capacity to buffer the payloads from eight packets, although those skilled in the art realize that the number eight is a design choice and might be varied without departing from the spirit of the invention. As packets arrive over asynchronous medium 103 the payloads are stored in buffers 500 sequentially, with wraparound; i.e., after the last of buffers 500 is used the next payload is stored into the first buffer. This is often referred to as a circular arrangement of buffers, or "a circular buffer". Those skilled in the art will recognize that other buffer arrangements may be employed. It is attempted to play data out of buffers 500 such that a buffer is played out before being overwritten by newly arriving data.

As indicated in Fig. 5 by the SEQ fields, arriving packets are be stored in buffers 500 in the order in which they were transmitted, said order being determined according to the SEQ fields. Thus, packets that are received out of order because of varying transmission delays are restored to correct order and the correct timing relationship among the packets may be determined. If the SEQ field contained in a buffer about to be played out indicates that the corresponding packet has not arrived, that packet is deemed to be lost in transmission, but the buffer is played out anyway with whatever content it happens to have. Thus, although message content is degraded with the loss in transmission of a packet, the relative timing of the payloads is unaffected. In an embodiment of the invention, should a packet that was deemed lost arrive after its corresponding buffer has been played out, it is simply discarded.

Fig. 6 is a block diagram of a system for manipulating the buffers 500 according to the present invention. Block 602 receives packets from asynchronous medium 103 and stores their payloads according to their SEQ fields in ones of buffers 500, using buffers 500 in a circular arrangement. Block 602 also maintains the arrival log 604. When operation of the system is initiated, the arrival rate is assumed to be equal to the transmission rate from the source PBX cabinet 101 (one packet every 500 μsec. in the present example). Thus local clock 606 is initialized according to the transmission rate from the source PBX cabinet 101. Block 610 plays out data from buffers 500, out of phase with the storing of buffers 500. That is, upon initialization of operation playing out does not begin until there is some backlog of data stored in the buffers 500.

Because of the asynchronous and possibly unreliable nature of communications medium 103, it is very unlikely that the actual arrival rate is precisely equal to the transmission rate from the source PBX cabinet 101, and thus it is unlikely that the initial value in clock 606 is exactly synchronous with the received data. Data loss is ameliorated by providing sufficient lag in buffers 500 to avoid loss of data during the time taken to correct the clock. Figs. 7, 8, and 9 show alternative systems for correcting local clock 606 according to arrival log 604.

In the system of Fig. 7, clock correction may be effected once each "check interval", (ChkInt). ChkInt should be chosen to be a multiple of the transmission period from the source PBX 103. In a present embodiment wherein the transmission period is 500μsec, ChkInt is chosen to be 50 msec. According to the system of Fig. 7, arrival log 604 is cleared at the beginning of a ChkInt, and then logs SEQ fields of packets received. In block 720, XmtCt is calculated, an estimate of the number of packets transmitted from source PBX 101 during ChkInt. XmtCt is constrained to being an estimate because of the asynchronous and unreliable nature of medium 103-- packets enroute to a destination PBX 101 might be buffered or otherwise delayed somewhere along medium 103. The determination of XmtCt takes into account any wraparound in the SEQ fields. For example, if SEQ fields take values

0-199, and if the highest and lowest SEQ fields recorded in arrival log 604 are 32 and 133 respectively, then XmtCt would reflect an estimate of 100 packets transmitted. (The range of SEQ fields is larger than the number of packets that could be transmitted during ChkInt.) XmtCt may differ from the number of packets received during ChkInt, in that some packets may have been lost in transit but SEQ fields of packets transmitted after them would be indicative that those lost packets had been transmitted.

Block 730 calculates DELTA by subtracting the number of packets that should have been transmitted during ChkInt from XmtCt. Said number of packets that should have been transmitted is determined as ChkInt divided by the transmit interval-- in the present example, 50 msec divided 500 μsec, or 100. Block 740 calculates an error indication e, and block 750 then adjusts local clock 606 accordingly. If XmtCt is higher than 100, clock 606 should be adjusted to play out packets more rapidly than the initially assumed 500 μsec interval; if lower than 100, packets should be played out more slowly. It is anticipated that in response to changing conditions on medium 103, both of these conditions will occur at different times in operation, and that at yet other times clock 606 will operate at the 500 μsec interval at which the source PBX 101 is transmitting. Inherent in the system of Fig. 7 is a tradeoff between the duration selected for ChkInt and the size of corrections that may be made to clock 606.

An alternative system for correcting clock 606 is depicted in Fig. 8. Again, a check interval connoted ChkInt is selected. Arrival log 604 is activated each ChkInt. Block 820 then notes the arrival time t_1 and the SEQ field value s_1 of the next packet received, connoted packet P_1. Block 830 then notes the arrival time t_2 and the SEQ field value s_2 of yet the next packet received, connoted packet P_2.

Block 840 checks that s_1 and s_2 are consecutive numbers (again, with accounting for possible wraparound) in order to determine whether packets p_1 and p_2 were transmitted sequentially. If not (indicating that one or more intervening packets may have been lost) control dispatches back to block 820 to make another try at receiving two sequentially transmitted packets; if so, control passes on to block 850.

Block 850 assess the current rate of change in delay on medium 103 as $(t_2 - t_1) - 500$ μsec (500 μsec being the interpacket interval of the packets as transmitted). Block 860 adjusts clock 606 according to the results of the value calculated in block 850. If the delay is currently constant, the time between arrivals of the two packets is equal to the 500 μsec period imposed by the source PBX 101, and the value calculated in block 850 is zero. If the delay is decreasing, packets appear to be arriving slightly more often than every 500μsec, and clock 606 should be speeded up accordingly. Conversely, if delay is increasing, packets appear to be arriving slightly less often than every 500 μsec and clock 606 should be slowed accordingly.

Under the system of Fig. 8, ChkInt may be selected as a much shorter interval than under the system of Fig. 7 without that system's effect of coarsening the granularity of correction. Thus, the rate of change in delay may be assessed often.

Fig. 9 depicts another alternative system for correcting clock 606. Here the rate of change of delay on medium 103 will be assessed over several packet times, specified as a number NumChk packet times. Block 920 sets a variable *n* to a value of 1. In block 930, the next packet is received and arrival log 604 stores its arrival time t_n and SEQ field value s_n.

In block 940, if at least two packets have been received a determination is made from the SEQ fields of the two most recently received packets whether an intervening packet has been lost. If a packet has been lost control returns to block 920 to initiate a new attempt to receive NumChk consecutive packets without a lost packet. If no packet has been lost control passes to block 950.

Block 950 determines whether NumChk packets have been received. If not, block 955 increments n and control passes to block 930 to continue accumulating packet data; if so, control passes to block 960.

Block 960 calculates the time between receipt of the first and last of the NumChk packets as $t_{rec} = t_{NumChk} - t_1$. Block 970 then calculates e, the difference between t_{rec} and the time that would have elapsed had delay on medium 103 been constant as $e = t_{rec} - (\text{NumChk}-1)*500\mu\text{sec}$. Block 980 then adjusts clock 606 according to the value of e.

An alternative scheme for reconstructing a local clock in a destination PBX cabinet 101 is depicted in Fig. 10. Block 702 receives packets from communications medium 103 and plays payload data into buffers 500. Block 702 also has a path for informing block 710 of the position within buffers 500 where it is presently storing data. Block 710 plays data out of buffers 500 at a rate determined by local clock 706 and from a position some predetermined increment behind the point at which block 702 is playing in data. That is, when operation is initialized, local clock 706 is initialized to the expected transmission rate, and play-out is not commenced until the aforementioned predetermined increment of data has been played in to the buffers 500. (For example, with buffering for the payloads from eight packets, the predetermined increment is one-half the buffer capacity, or four payloads.)

Block 710 reports to block 708 the positions of play-in and play-out. If play-out is ahead by less than the predetermined increment, block 708 adjusts local clock 706 so as to speed it up. Conversely, if play-out is ahead of play-in by more than the predetermined increment, block 708 adjusts local clock 706 so as to slow it down. Local clock 706 may be used for conventional timing functions relative to processing incoming data (e.g., delivering audio to the earpiece of a phone), as was the case with local clock 606 of Fig. 6.

The functions depicted in Figs. 6 through 10 may be performed in an arithmetic and logic unit (ALU), which may be generically described as a digital device responsive to preprogrammed instructions. In a present embodiment, the reception of packets from communications medium 103 and the play-in and play-out of buffers 500 are handled by a programmable gate array (PGA) responsive to instructions coded in the C programming language, and the other functions described are performed in a microprocessor already extant in a PBX cabinet. Those skilled in the art appreciate that a wide variety of microprocessors, PGA's, application-specific integrated circuits (ASIC's), and reduced instruction-set controllers (RISC's), etc. may be employed for the purpose.

It will thus be seen that the invention attains the advantages set forth above, among those made apparent from the preceding description. In particular, the invention provides reconstruction of a local clock for synchronous data after asynchronous transmission. Those skilled in the art will appreciate that the buffering depicted in Fig. 5 in conjunction with the play-out and clock adjustment systems depicted in Figs. 6 through 10 allows the transmission speed of data over an asynchronous medium to vary, and allows reconstruction of a clock at the receiving end. The clock thus reconstructed varies in speed so as to re-establish synchrony with the data.

It will be understood that changes may be made in the above construction and in the foregoing sequences of operation without departing from the scope of the invention. It is accordingly intended that all matter contained in the above description or shown in the accompanying drawings be interpreted as illustrative rather than in a limiting sense.

It is also to be understood that the following claims are intended to cover all of the generic and specific features of the invention as described herein, and all statements of the scope of the invention which, as a matter of language, might be said to fall therebetween.

Having described the invention, what is claimed as new and secured by Letters Patent is: